Case Study: American Religious Ecologies

by Greta Swain

Introduction 1

About the Project 2

About the Data 3

Overview of Transformation Process 4

Digitization, Metadata Creation and Omeka Import 4

Digitization 5

Metadata 5

Omeka Setup and Import 6

Transcription and Dataset Creation with DataScribe 7

Transcription Before DataScribe 8

Transition to using DataScribe 9

Creating the Transcription Form 9

Using DataScribe for Project Management 12

Workflow 12

Assigning Work 13

Transcribers and Reviewers 13

Deploying DataScribe for Research 14

Dataset Export 14

Research and Visualizations 14

Conclusion 15

Introduction

Are you curious about using DataScribe for your own data transcription project but would like to see how it's worked out for another project first? Perhaps you have hundreds (or thousands!) of documents that you've photographed sitting on your hard drive, but you're not sure what to do next. Or maybe you already have a collection of sources in Omeka S and would like to transcribe them within your current infrastructure. Or it is possible that you're in the middle of transcribing your sources into a spreadsheet, and you're hesitant to make a change mid-project, so you'd like to better understand what using DataScribe would entail.

In the following case study we describe how the American Religious Ecologies project at the Roy Rosenzweig Center for History and New Media uses DataScribe to transcribe tabular data from early twentieth-century digitized census forms in order to create a new dataset for American religious history. You can see the forms (called "schedules") from the 1926 U.S. Census of Religious Bodies that we are transcribing at our Omeka S website.

Because DataScribe was not yet created when our team started transcription, we first used a basic shared spreadsheet for transcription and transitioned to DataScribe early in the project. In this study, we will provide an overview of the American Religious Ecologies project and the sources we used. We will also detail the process of digitizing and readying our sources for transcription, the transition from using spreadsheets to using DataScribe for transcription, the workflows we developed for transcribing and reviewing, DataScribe as a project management tool, the final format of the data coming out of DataScribe, and the questions and visualizations this new dataset enabled. Finally, we discuss the decisions we made along the way. This case study will give you a better sense of how DataScribe was used by a large team at a university research center to asynchronously manage a large-scale transcription project. DataScribe is a critical part of our efforts to transcribe hundreds of thousands of pages of forms and create new datasets.

About the Project

American Religious Ecologies is a project of the Roy Rosenzweig Center for History and New Media. The project aims to create new datasets, maps, and visualizations for the history of American religion. The Religious Ecologies team is fairly large: over forty people have been connected with the project so far. American religious historians work alongside developers, designers, graduate research assistants, undergraduate research assistants, and an eight member advisory board. The project regularly has upwards of a dozen team members working at any one time, many of them undergraduate research assistants.

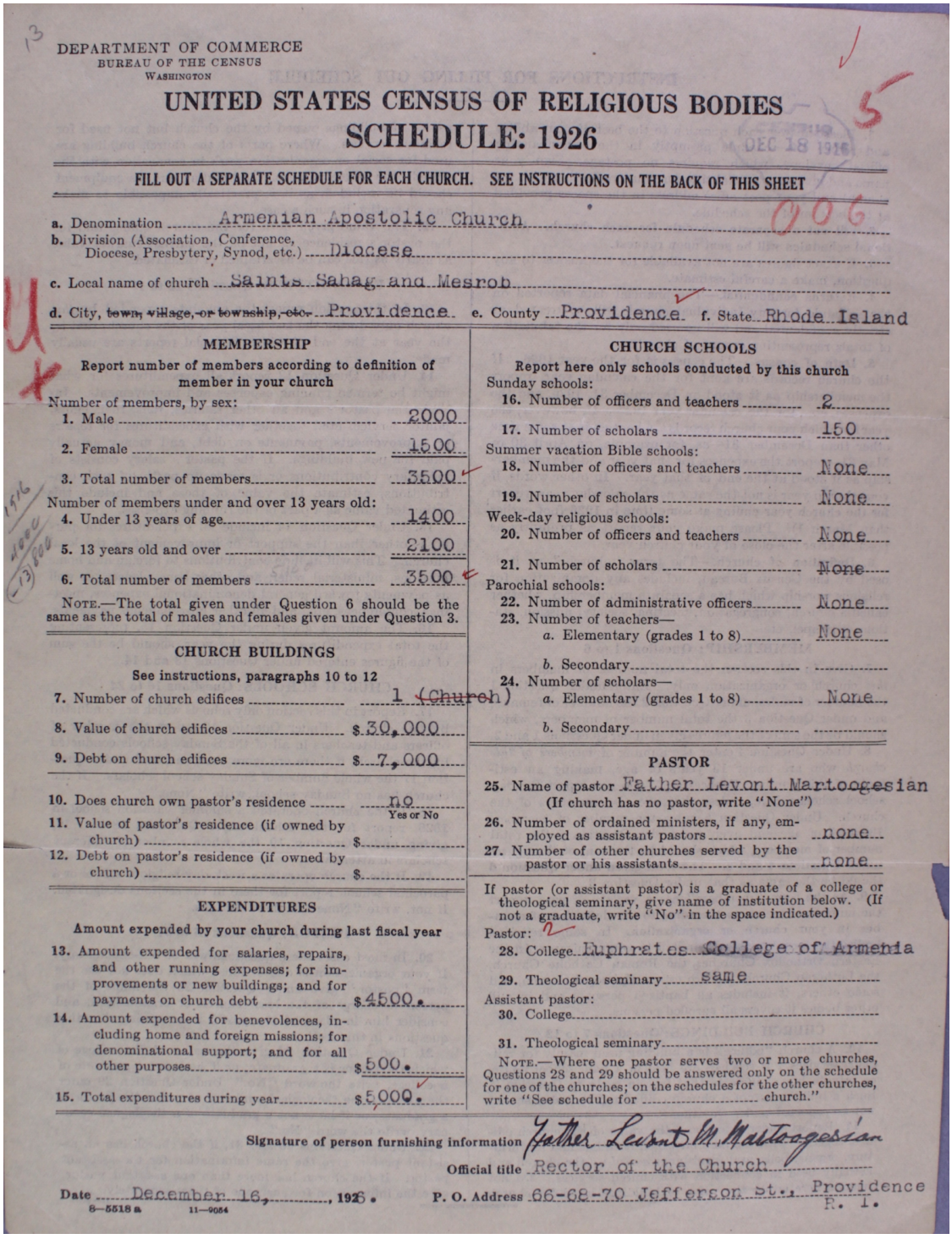
In 2018, the project received funding from the Division of Preservation and Access at the National Endowment for the Humanities to digitize and transcribe records from the 1926 U.S. Census of Religious Bodies. Since then, we have undertaken a number of steps to turn those records into a dataset: digitizing approximately 232,000 census schedules held the National Archives and Records Administration, creating basic metadata for each document, importing the documents into the content management tool Omeka S, using DataScribe to transcribe data from the census forms, constructing spatially-linked datasets, developing data-driven visualizations, and presenting our early findings in blog posts (https://religiousecologies.org/blog/) and conference presentations.

About the Data

For the past several years, the *American Religious Ecologies* project has been digitizing and transcribing approximately 232,000 “schedules” from the 1926 U.S. Census of Religious Bodies. Many are familiar with the U.S. Census Bureau because of the decennial population census. However, in the late nineteenth and early twentieth centuries, the Census Bureau also collected information about religion in the United States. In 1902, Congress established the Census Bureau as a permanent office, and authorized them to undertake a separate decennial census of “religious bodies”—which they attempted in 1906, 1916, 1926, 1936, and 1946. This census, like the census on manufacturing, was separate from the population census, counting institutions rather than individuals. Congregations from across the nation were provided with a one-page form or “schedule” where they supplied information about their location, denomination, membership, finances, Sunday schools, and clerical staff.

After the schedules were returned, the Census Bureau counted and summarized the information using punch-card tabulating machines. (In a sense, our project is “re-digitizing” the census, because the Bureau’s punch cards were already a digital representation.) The Bureau published aggregated data by city and denomination in large volumes and the individual schedules were destroyed. However, the 1926 schedules somehow survived extermination, and they are now housed in 314 archival boxes at the National Archives and Records Administration in downtown Washington, D.C.

The 1926 Religious Bodies schedules tell us a great deal about individual congregations, but also about how the Census Bureau viewed American religion. First of all, the Census Bureau created a standard form that was intended to be filled out by a single church, congregation, or parish. There were some variations on that form, including a separate schedule for Jewish congregations, but in general the Bureau thought of religious groups as fitting into a Protestant, congregational model.



DEPARTMENT OF COMMERCE
BUREAU OF THE CENSUS
WASHINGTON

UNITED STATES CENSUS OF RELIGIOUS BODIES
SCHEDULE: 1926

FILL OUT A SEPARATE SCHEDULE FOR EACH CHURCH. SEE INSTRUCTIONS ON THE BACK OF THIS SHEET

a. Denomination: Armenian Apostolic Church
b. Division (Association, Conference, Diocese, Presbytery, Synod, etc.): Diocese
c. Local name of church: Saints Sahag and Mesrob
d. City, ~~town, village, or township, etc.~~: Providence e. County: Providence f. State: Rhode Island

MEMBERSHIP

Report number of members according to definition of member in your church

Number of members, by sex:
1. Male: 2000
2. Female: 1500
3. Total number of members: 3500

Number of members under and over 13 years old:
4. Under 13 years of age: 1400
5. 13 years old and over: 2100
6. Total number of members: 3500

NOTE.—The total given under Question 6 should be the same as the total of males and females given under Question 3.

CHURCH BUILDINGS

See instructions, paragraphs 10 to 12

7. Number of church edifices: 1 (Church)
8. Value of church edifices: $30,000
9. Debt on church edifices: $7,000
10. Does church own pastor's residence: no (Yes or No)
11. Value of pastor's residence (if owned by church): $
12. Debt on pastor's residence (if owned by church): $

EXPENDITURES

Amount expended by your church during last fiscal year

13. Amount expended for salaries, repairs, and other running expenses; for improvements or new buildings; and for payments on church debt: $4500.
14. Amount expended for benevolences, including home and foreign missions; for denominational support; and for all other purposes: $500.
15. Total expenditures during year: $5000.

CHURCH SCHOOLS

Report here only schools conducted by this church

Sunday schools:
16. Number of officers and teachers: 2
17. Number of scholars: 150

Summer vacation Bible schools:
18. Number of officers and teachers: None
19. Number of scholars: None

Week-day religious schools:
20. Number of officers and teachers: None
21. Number of scholars: None

Parochial schools:
22. Number of administrative officers: None
23. Number of teachers—
a. Elementary (grades 1 to 8): None
b. Secondary:
24. Number of scholars—
a. Elementary (grades 1 to 8): None
b. Secondary:

PASTOR

25. Name of pastor: Father Levont Martoogesian (If church has no pastor, write "None")
26. Number of ordained ministers, if any, employed as assistant pastors: none
27. Number of other churches served by the pastor or his assistants: none

If pastor (or assistant pastor) is a graduate of a college or theological seminary, give name of institution below. (If not a graduate, write "No" in the space indicated.)

Pastor:
28. College: Euphrates College of Armenia
29. Theological seminary: same

Assistant pastor:
30. College:
31. Theological seminary:

NOTE.—Where one pastor serves two or more churches, Questions 28 and 29 should be answered only on the schedule for one of the churches; on the schedules for the other churches, write "See schedule for ____ church."

Signature of person furnishing information: Father Levont M. Martoogesian
Official title: Rector of the Church
Date: December 16, 1926
P. O. Address: 66-68-70 Jefferson St., Providence R. I.

8—5518 a 11—9054

Figure 1. An example of a schedule from the 1926 Census of Religious Bodies. This schedule was filled out by an Armenian Apostolic Church in Providence, Rhode Island. You can read more about this schedule in a blog post on "What can you learn from a census schedule?"

Churches indicated the denomination and division (association, conference, diocese, presbytery, synod, etc.) to which they belonged. The standard form worked well for most Protestant groups, but it was a poor fit for others. Even Protestant churches often wrote additional comments on the schedules because they felt the form did not suit their particular circumstances.

Second, the Census Bureau asked for geographic information such as the church's city, county, state, and an address where the person furnishing information could be reached. These geographical fields have helped our team sort congregations by location and will be the basis of our mapping efforts. Also at the top of the forms you'll see red pencil and ink markings made by the Census Bureau as they assigned each schedule an ID number and a three-digit denominational ID, which we use to sort the schedules by denomination.

Finally, it is important to note that the Census Bureau asked questions about (and therefore ascribed meaning to) aspects of religion that could be easily counted on punch cards and summarized. Therefore, the schedule is filled with many quantitative questions such as the number of members, the value of the church's edifice, its total financial expenditures, the number of Sunday schools, and number of assistant pastors.

For more detailed information about the schedules, see our blog post about what you can learn from a census schedule. Our blog also has many "schedule spotlights" that uncover the history of specific congregations using the census schedules.

How Sources Are Transformed into Data

The Census Bureau's form was standardized, and much of the information can be thought of as "strongly typed." In other words, fields can be recorded as integers, other numeric data, locations, IDs, and categorical data. Moreover, each field on the schedules contains only a single entry, so no complicated, relational data structure is necessary to accurately represent the structure of the data. In essence, you can think of the census data, once transcribed, as a giant table of data. There is one column (a variable) for each of the fields on the census schedule. And there is one row (an observation) for each congregation that filled out a schedule. This tabular structure makes the 1926 Religious Bodies schedules a prime candidate for structured data transcription with DataScribe.

Over the past few years, two endeavors have been central to the American Religious Ecologies project: (A) photographing approximately 232,000 schedules from the 1926 Census of Religious Bodies in order to make them freely available and searchable online, and (B) creating spatially-linked datasets from those schedules that tell us about American religious life in the early twentieth century.

This is a complex process to get from point A (physical documents in the archive) to point B (a dataset that can be computationally analyzed and visualized).

First, we created digital copies of our sources by taking photographs of the documents of interest. Next we created metadata to describe each document, and imported the metadata and document images into Omeka S, a content management system developed by RRCHNM. Then we made these schedules freely available on our 1926 Census of Religious Bodies website, where users can browse the schedules by location or denomination, or explore them through a map interface. Finally, we used DataScribe—a new module for transcribing structured data in Omeka S—to transcribe the data from the Census forms and transform it into a dataset.

Conceptually, this final process of extracting information found in historical sources and converting it into a dataset is is really two-fold: first, one must transcribe text or numbers from the document, or in this case, type a copy of the text and numbers into the computer so that they become machine-readable. Second, one must organize and give structure to this text in order to transform it into data. For many years, historians have been interested in sources—like the 1926 census schedules we are digitizing—which contain quantitative information or statistics. To record these details and make use of them, scholars have often utilized spreadsheet software that only really helped with the first part of the process—-transcribing the text or numerals; they did not help keep the information uniform or give it structure. DataScribe lets us both transcribe and standardize our data—in effect allowing us to undertake both of the steps in the transformation process at one time.

Digitization, Metadata Creation and Omeka Import

Transforming archival sources into digitized documents that can be viewed and searched in an online content management system like Omeka S is a muti-step process. We broke this down into several major steps: (1) digitization of census schedules, (2) metadata creation, and (3) importing the schedules into Omeka S.

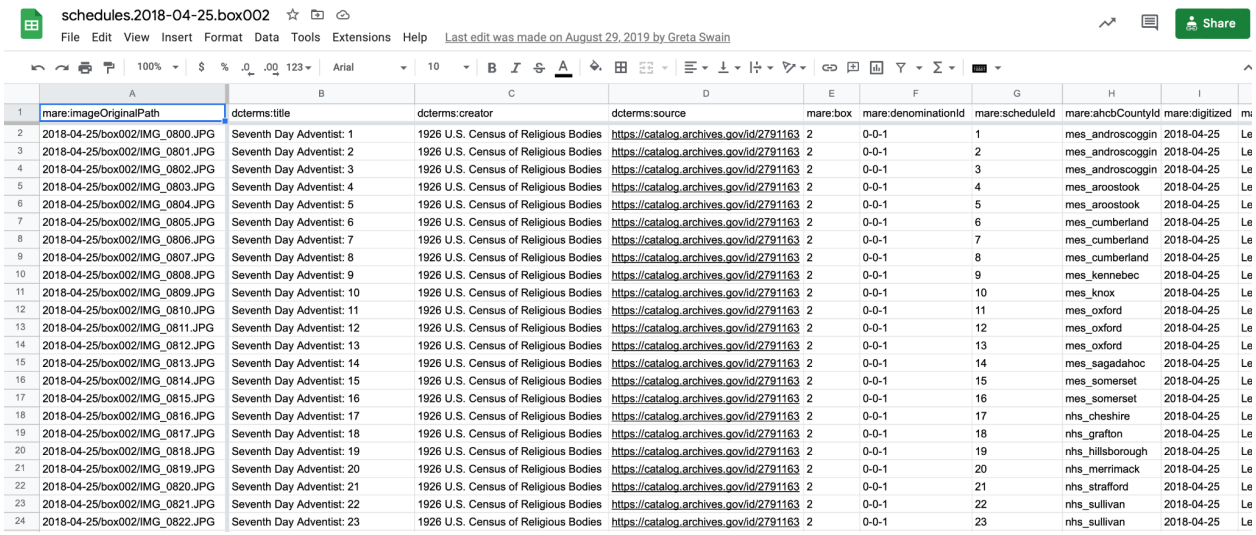
Digitization

We digitized the schedules from the 1926 Census of Religious Bodies by sending project team members to the National Archives Research Center in Washington, DC, to take high-quality photographs of approximately 232,000 schedules. We used a photo stand provided at the archives to ensure that our images were uniform and properly lit. We also set the camera to save both a RAW and a JPEG version of each image; this ensured that we captured the highest quality image possible (recording all of the data from the camera's sensor) while also obtaining an image in a more usable format. We stored the photographs on a server and organized them by the date photographed and

the archival box number. We collaboratively keep track of our progress in a Google Spreadsheet. Next, we used a short command-line computer program to automatically crop and rotate the images.

Metadata

After that, in a task we called “cataloging,” we worked to create metadata—data which describes or gives information about other data—for each schedule. We divided the images into “batches" or groups based on their arrangement on the server, and recorded the metadata for each batch on a separate spreadsheet. We asked our undergraduate and graduate research assistants to record several different types of metadata. We duplicated some of the original metadata that the Census Bureau created in the 1920s (e.g., an ID number for each schedule and an ID code for each denomination). We recorded geographic information such as the state and county where the religious congregation was located. We included information about the original document (e.g., the archival box number of the schedule’s current location at NARA) and the digital copy we made (e.g., the date the schedule was photographed and the name of the photographer). Finally, we included some administrative metadata, such as the image’s location on the server.



schedules.2018-04-25.box002

File Edit View Insert Format Data Tools Extensions Help Last edit was made on August 29, 2019 by Greta Swain

Share

	A	B	C	D	E	F	G	H	I
1	mare:imageOriginalPath	dcterms:title	dcterms:creator	dcterms:source	mare:box	mare:denominationId	mare:scheduleId	mare:ahcbCountyId	mare:digitized
2	2018-04-25/box002/IMG_0800.JPG	Seventh Day Adventist: 1	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	1	mes_androscoggin	2018-04-25
3	2018-04-25/box002/IMG_0801.JPG	Seventh Day Adventist: 2	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	2	mes_androscoggin	2018-04-25
4	2018-04-25/box002/IMG_0802.JPG	Seventh Day Adventist: 3	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	3	mes_androscoggin	2018-04-25
5	2018-04-25/box002/IMG_0803.JPG	Seventh Day Adventist: 4	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	4	mes_aroostook	2018-04-25
6	2018-04-25/box002/IMG_0804.JPG	Seventh Day Adventist: 5	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	5	mes_aroostook	2018-04-25
7	2018-04-25/box002/IMG_0805.JPG	Seventh Day Adventist: 6	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	6	mes_cumberland	2018-04-25
8	2018-04-25/box002/IMG_0806.JPG	Seventh Day Adventist: 7	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	7	mes_cumberland	2018-04-25
9	2018-04-25/box002/IMG_0807.JPG	Seventh Day Adventist: 8	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	8	mes_cumberland	2018-04-25
10	2018-04-25/box002/IMG_0808.JPG	Seventh Day Adventist: 9	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	9	mes_kennebec	2018-04-25
11	2018-04-25/box002/IMG_0809.JPG	Seventh Day Adventist: 10	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	10	mes_knox	2018-04-25
12	2018-04-25/box002/IMG_0810.JPG	Seventh Day Adventist: 11	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	11	mes_oxford	2018-04-25
13	2018-04-25/box002/IMG_0811.JPG	Seventh Day Adventist: 12	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	12	mes_oxford	2018-04-25
14	2018-04-25/box002/IMG_0812.JPG	Seventh Day Adventist: 13	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	13	mes_oxford	2018-04-25
15	2018-04-25/box002/IMG_0813.JPG	Seventh Day Adventist: 14	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	14	mes_sagadahoc	2018-04-25
16	2018-04-25/box002/IMG_0814.JPG	Seventh Day Adventist: 15	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	15	mes_somerset	2018-04-25
17	2018-04-25/box002/IMG_0815.JPG	Seventh Day Adventist: 16	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	16	mes_somerset	2018-04-25
18	2018-04-25/box002/IMG_0816.JPG	Seventh Day Adventist: 17	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	17	nhs_cheshire	2018-04-25
19	2018-04-25/box002/IMG_0817.JPG	Seventh Day Adventist: 18	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	18	nhs_grafton	2018-04-25
20	2018-04-25/box002/IMG_0818.JPG	Seventh Day Adventist: 19	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	19	nhs_hillsborough	2018-04-25
21	2018-04-25/box002/IMG_0819.JPG	Seventh Day Adventist: 20	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	20	nhs_merrimack	2018-04-25
22	2018-04-25/box002/IMG_0820.JPG	Seventh Day Adventist: 21	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	21	nhs_strafford	2018-04-25
23	2018-04-25/box002/IMG_0821.JPG	Seventh Day Adventist: 22	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	22	nhs_sullivan	2018-04-25
24	2018-04-25/box002/IMG_0822.JPG	Seventh Day Adventist: 23	1926 U.S. Census of Religious Bodies	https://catalog.archives.gov/id/2791163	2	0-0-1	23	nhs_sullivan	2018-04-25

Figure 2. An example of a spreadsheet that catalogs the digitized schedules. The first column contains the path to the image to be imported, and the rest of the columns contain metadata about the image or the schedule which will be stored as fields on the Omeka S item.

By associating an image of each schedule with machine-readable text that described the schedule, we prepared our digitized schedules to be searchable by the different fields we recorded. For example, because we recorded metadata like the

denomination ID and congregation's location, visitors to our project's website can now sort the schedule by denomination (e.g., find all the congregations that belonged to the Advent Christian Church) or location (e.g., find all the congregations located in Suffolk County, Massachusetts).

In order to create our metadata in the right format, we had to anticipate how we might carry out the Omeka import process. Early on, we decided that the Omeka CSV Import plugin was the best way to import a large number of schedules and their accompanying metadata into Omeka S. This choice shaped how we structured the metadata and how we addressed irregularities in the data. For example, we wanted each church or congregation to be recorded as a separate Omeka item. Since, the plugin creates one Omeka item for each row in the spreadsheet, we made sure to record the metadata for each schedule in a single row. We also organized the metadata by fields, representing each category of metadata (schedule ID, date digitized, etc.) in its own column. This ensured a smooth mapping (matching) between our columns and the metadata fields in the Omeka S import process.

However, even with our careful planning, we ran into some irregularities in our data that caused us to adapt our original plans. At first, we thought that each individual schedule form had only one schedule ID and represented one congregation. But a few weeks into our digitization efforts, we came across a single schedule with six schedule IDs at the top, representing six separate congregations. This led us to reassess our earlier assumptions that each schedule (an individual piece of paper) would be represented by only one row. Instead, we needed to create six rows in the spreadsheet (one row per schedule ID or congregation) in order to represent all the congregations found on a single schedule. In other words, we realized that a "congregation" (represented by one schedule ID) instead of a "schedule" formed our basic unit of analysis.

Omeka Setup and Import

Finally our last step was to import our schedules into Omeka S in batches, using Omeka's CSV import module. We customized our import by adjusting some settings and options. First of all, we created a custom resource template in Omeka S for our schedules. There we specified the metadata fields that we wanted Omeka to create for each congregational item; these mirrored the metadata fields (columns) in our cataloging spreadsheets. We also created item sets (or collection of items)—one for all of the items pertaining to the 1926 Census of Religious Bodies, and another just for census schedules. During the Omeka import (under "Basic Settings"), we were able to select our custom resource template, and both of our item sets to ensure that the new batch being imported would use our custom template, and be added to both of the item sets we created. Having an item set with all of the schedules in it became particularly important when we started using DataScribe for our transcription.

During the import, we made adjustments under the “Map to Omeka S data” tab to ensure that columns in the spreadsheet we were importing matched up to the Omeka metadata fields we had created. We specified that our first column (representing the schedule’s image file) was a media source, and through the “sideload” option (enabled by the File Sideload module), had Omeka find the corresponding schedule image on the server, and attach a copy of it to the Omeka item for each congregation. For our “source” field, we wanted to provide a link to the National Archives’ webpage about the 1926 Census of Religious Bodies. Using the “Configure column” option, we set the data type for this column to be an URI (uniform resource identifier) instead of text. This meant that for each Omeka item, the information in the “Source” field showed up as a clickable link. In total, the CSV import process had created one Omeka item for each row in our spreadsheet, and attached to this item both the metadata we had created and the photograph of the schedule we had taken.

At this point in the process, we had achieved one of our goals: photographing schedules from the 1926 Census of Religious Bodies and making them publicly available online. However we still had an important goal remaining: transforming these digitized schedules into a dataset.

Transcription and Dataset Creation with DataScribe

As mentioned earlier, the process of taking the information found in historical sources and transforming it into a machine-readable dataset is really two-fold. First, there is the basic step of transcription—-writing or typing a copy of the numbers or text found on the document. For scholars wanting a transcription of a prose document for non-data purposes, they can end their transcription process after this first step. But for scholars with tabular or quantitative data, or for those wanting to use their sources to create datasets that can be computationally analyzed or visualized, a second step of organizing and structuring the data is necessary.

First of all, documents that are filled out by humans—like our 1926 census schedules—naturally have small variations in them. In other words, congregations represented the information they furnished to the Census Bureau in different ways. For example, some congregations used text to record that they had “one” church building, while others represented this data with a numeral “1” instead. When looking at a few documents, these variations seem insignificant; a human mind can quickly determine that “one” and “1” have the same meaning. But when you want to ask a computer to analyze or visualize thousands of documents by looking at a single variable—say number of church buildings—these variations become more problematic. This is one reason why it is so important that the data for datasets is transcribed in a uniform way.

Another issue is that sources sometimes do not have clearly defined fields or they have text that is subject to interpretation. For example, the Census Bureau asked questions about the pastorate of each congregation: What was the pastor’s name?

Were they a college graduate, and if so, from what school? Were they a graduate of a theological seminary, and if so, from what school? While some congregations simply wrote the names of the schools their pastors had graduated from (e.g., “Cornell”, “Princeton”), other congregations explained that their pastor had attended 2 years at one school but did not have a degree. So our job was to figure out how to best interpret and record this. Was it important to keep track of all the names of colleges where pastors had attended? Or did we throw out this piece of data (name of school pastor attended for two years) because we were only interested in the names of schools from which pastors graduated? Did we need to create a categorical variable to identify that a pastor “only attended” vs. “graduated” from a particular school? This type of messy data which requires interpretation, categorization and decision making is a far cry from the cut and dry data of “1” church building, and is often hard to keep track of in a basic spreadsheet.

DataScribe’s ability to help us both transcribe and standardize our data in one step made it the perfect tool for creating a dataset from the 1926 Census of Religious Bodies schedules. In addition, because DataScribe is built on the same Omeka S platform that we were already using to store our schedule images and metadata, it was easy to synchronize our data across modules. The following sections will give details about how we transitioned from using a standard spreadsheet for our data transcription to using DataScribe, and explain how we set up and customized the DataScribe module in Omeka S to fit our transcription project needs.

Transcription Before DataScribe

During the early stages of the *American Religious Ecologies* project, DataScribe was not yet under development. This meant that our project started transcribing the census schedules using basic spreadsheets before later transitioning our transcription work to DataScribe. We set up our transcription spreadsheets much like the ones we used for metadata creation: we created one column for every field listed on the schedule (name of the local church, total number of members, total expenditures, name of pastor, etc.) and one row for every congregation. In an attempt to try to standardize the data, we created a long list of rules that dictated how transcribers should interpret and transcribe the various entries for each field. For example, if a field was left blank, transcribers were to enter “NA” into the spreadsheet cell to indicate a null field. If a number was given, whether in textual or numeric form, they were instructed to enter it as a number. If the congregation wrote “none,” transcribers were asked to enter this as “0.” After every data field, we also included an accompanying “flag” column. If the transcribers were unsure of what to enter for a particular field, we asked them to leave the data column for that field blank, and instead enter the word “TRUE” in the flag column so we could easily identify which items needed further attention.

Overall, this process was very cumbersome and confusing, and it was easy for

transcribers to make a mistake, forget what the transcription rules dictated, or include a typo. In addition, if we had kept transcribing only in the spreadsheet, we would have needed to do a lot more work at the end of our process to make the data usable for computational analysis. For example, all the data in the spreadsheet—whether numbers or text—would have automatically been interpreted by a computer as “characters.” In order to compute totals or averages of numbers, or computationally analyze true/false answers, it would have been necessary to first parse and validate the data—essentially converting each field into the correct data type (character, integer, boolean, etc.) and checking it for accuracy.

Transition to using DataScribe

After the beta version of DataScribe was released in November 2020, we transitioned our transcription to DataScribe. First, we installed the DataScribe module in our existing Omeka S installation (where all of our congregational Omeka items were located). Then, we created a DataScribe “project” which we called “Religious Ecologies” that served as the hub or container for all the transcription projects we wanted to undertake now or in the future for this project. Within this project, we created a dataset for our “1926 Schedules Transcription.” (In the future, our project might want to create other datasets using different sources, so those could be added as new datasets within our same “Religious Ecologies” project.) While creating a new dataset, you choose an item set or collection of items from your Omeka S install that you would like to transcribe. As mentioned previously, we had already created an item set called “Schedules” and had automatically added all schedules to it during the import process; we chose this set of items for our “1926 Schedules Transcription” dataset.

Next we used the “sync dataset” feature to move copies of items from our selected Omeka S item set into the DataScribe interface. However, we did not have all 232,000+ census schedules imported into Omeka S before we started our work in DataScribe. Instead, we continued to import schedules into Omeka S and to our “Schedules” item set over time. Therefore, while we used the “sync dataset” feature to initially bring our first round of schedules into DataScribe, we also used it to add copies of newly imported schedules to DataScribe as we went along.

Creating the Transcription Form

One of the ways that DataScribe helps increase the structure, accuracy, and usability of transcriptions is by having transcribers fill out a consistent, pre-determined form for each source. DataScribe includes a form builder which makes it easy to create a custom form that is tailored to your sources and data needs. The form builder offers a variety of different field types to choose from, including text boxes for textual entries, date, time and number fields, check boxes, radio buttons, and dropdown menus for categorical variables. When building your form, you can include transcription

instructions or input tables that will be shown in the transcription interface, dictate what type of data is acceptable for a transcriber to enter (including minimum and maximum length or data type), insert placeholder text, decide if specific fields are required or not, and in some cases, offer concrete choices to transcribers. For example, we were able to pre-determine that membership statistics should only be entered as numbers (instead of text) and disallow any other type of input for those fields. For each field, DataScribe also provides check boxes to allow transcribers to mark if the information in the field is missing or is illegible. This replaced our earlier use of a “flag” column.

While the form builder is easy to use, creating your form takes a lot of deliberate thinking and planning; the choices made at this stage affect the rest of your transcription efforts and determine the shape of your dataset in the end. We began building our form by creating one field for each question asked on the census schedules. For questions where congregations had written in textual responses (church name, pastor name, etc.) we used the text field type. This recorded data in a format that the computer recognizes as “characters.” For responses that were numerical (total number of members, church expenses in dollars, etc.), we used a number field type; this resulted in data stored as integers. For questions with concrete options, (Does the congregation own a pastors’ residence? yes or no; Is this an urban or rural congregation?), we chose to use field types like radio (which creates radio buttons) and select (which creates a dropdown menu of options). We could have, of course, used a text field for this data, allowing transcribers to type “Yes” or “No,” “Urban” or “Rural.” However, by doing this, we would have given transcribers the opportunity to create typos or record their answers in different ways. For example, we might have seen “yes” recorded as: yes, YES, yEs, or Yes. And while as humans, we know that those all mean the same thing, it is tricker for a computer to know that. So by using radio buttons or a dropdown of options, we were able to give transcribers concrete options and knew that the computer would record them consistently.

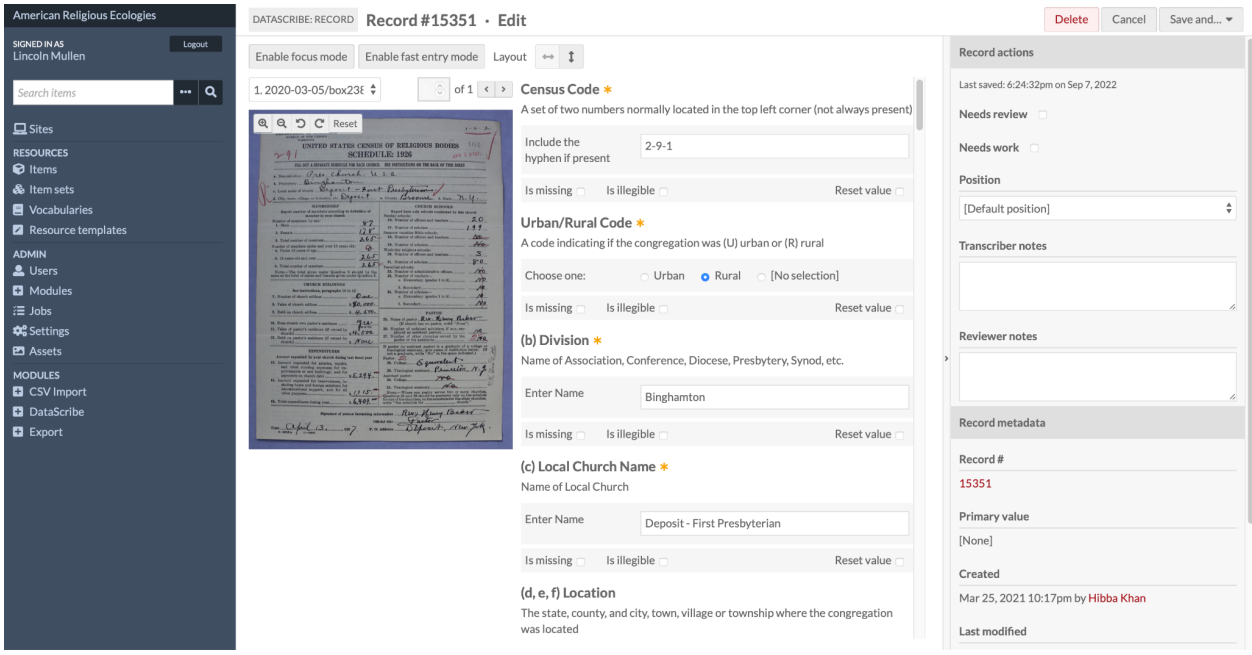


Figure 3. The DataScribe transcription form for American Religious Ecologies, showing an example of a schedule being transcribed.

Another key piece of information on each schedule was the congregation's location. We knew we wanted to create maps showing the locations of congregations, so recording this data in a usable format was a crucial step of the transcription process. On the schedules themselves, this location information was recorded in three separate fields: (1) state, (2) county, and (3) city, town, village or township. This is, of course, inherently spatial data. But if we simply transcribed this location information as text in several text fields, there would be no connection between the place name and its geospatial location. Of course, we could have gone back at the end and tried to geolocate the city or town recorded on 230,000+ schedules, but that would have been cumbersome and extremely time consuming.

Our solution was to enlist DataScribe in helping us create connections between the textual place names and their geospatial locations as we transcribed the schedules, instead of at the end of transcription. We accomplished this by creating a custom field type—which we called "Populated Place"—which draws on a spatial dataset of almost 200,000 U.S. place names. Instead of typing the state, county or city name, this field lets a transcriber choose the congregation's location from a series of nested dropdown menus. Transcribers first select a state from a dropdown menu. They are then presented with a second dropdown that lists all the counties in that state. After selecting the proper county, transcribers see a third dropdown that lists all the populated places (cities, towns, etc.) in that county. When DataScribe saves this input, it does not record the text of that place name; instead it records a six-digit spatial code that helps link that populated place to specific latitude and longitude coordinates. Having these geospatial

codes saved as part of the dataset made it much easier to start digitally mapping the locations of these congregations, even part way through the transcription process.

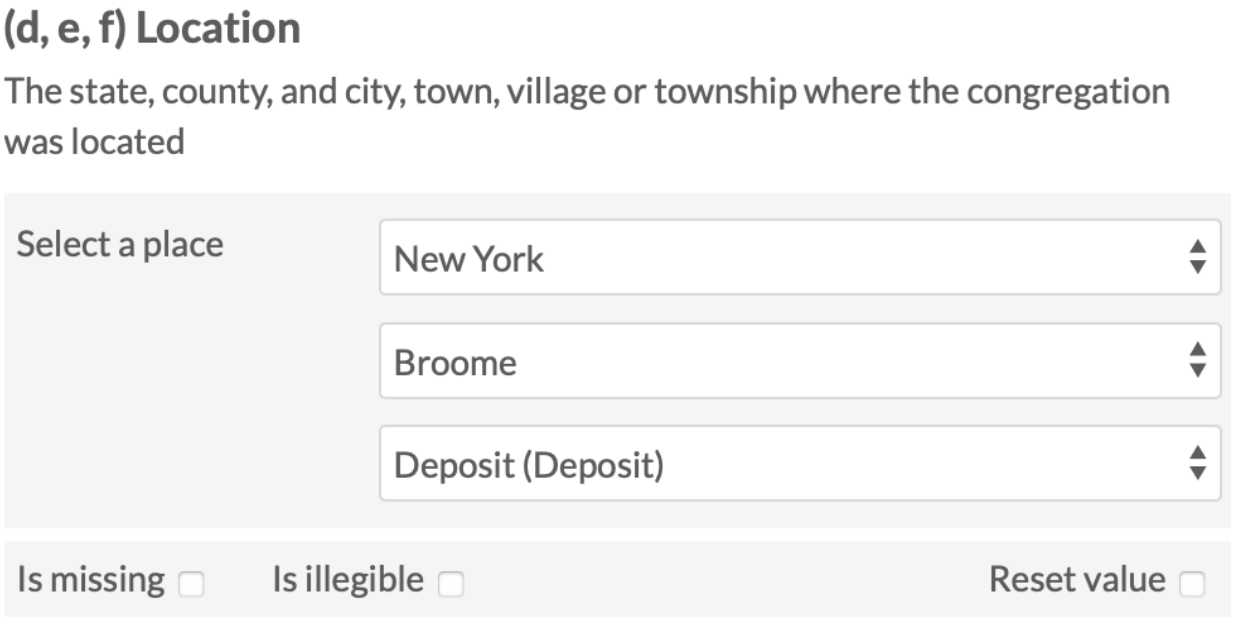


Figure 4. The transcription of a populated place allows the transcriber to select the state, county, and name of the populated place from a dropdown. This custom functionality was added to the project because DataScribe is itself extensible.

Of course, we realized that it was entirely possible that a congregation might have recorded a place name that is no longer a place, or was not included in our dropdown menu. Because of this, we included a text field called “Unlisted Populated Place” directly below our other location field. This gave transcribers a text box to record a place name that could not be found through the first method. We instructed transcribers to only use this second field if they could not find the location through the first method, and to leave this field blank if they were successful in using the populated place field. We found that this secondary field was only needed on a small number of schedules, and we were able to easily hand-geocode these few locations later on.

Another dilemma we encountered when creating our form was deciding how to ask transcribers to record information from the census schedule that did not fit into our “one question equals one field” convention. Some of the questions asked by the census bureau were really two questions in one. For example, for question 25, they asked for the name of the pastor, but also indicated that “If church has no pastor, write ‘None’.” In reality, this question was asking: (1) does this church have a pastor, and if they do, (2) what is the pastor’s name? By recording this information in a single text field, we would muddy the distinction between these two questions. So instead, we created two fields on the DataScribe form—a radio field for the yes/no question and a text field for the

pastor's name. We included instructions for transcribers to mark the second field as "is missing" if the congregation did not have a pastor.

After creating all of our fields and customizing them to fit our needs, we were ready to start transcribing. But before our DataScribe transcription got underway, we wanted to make sure that the transcription work we had already completed did not go to waste. To do this, one of the developers on our team wrote a script to help import the transcriptions that we had already completed in the spreadsheet into DataScribe, mapping the columns from the spreadsheet to the fields we had created. This process created new transcription records for the corresponding DataScribe items. We were then able to review our previous work within the DataScribe interface. We also had to go through each imported schedule individually to create a geospatial code through the populated place field for each imported transcription.

Overall, creating (and then using) DataScribes's custom transcription form helped us standardize our data, kept transcription errors to a minimum, and made each field easier to analyze or visualize.

Using DataScribe for Project Management

In addition to helping us organize our data, DataScribe also helped us organize and manage our project. When you are working on a transcription project by yourself, or with one or two other people, or if you are transcribing a small amount of sources, it might be plausible to keep track of your progress on your own. But for the American Religious Ecologies team—a large project team working and communicating asynchronously—this would have been more difficult without dedicated software or features. DataScribe made it easy for our team—transcribers, reviewers and our project manager—to collaborate on the transcription process. DataScribe allowed us to give team members specific roles (transcriber or reviewer), let our project manager assign tasks and review transcription progress, and gave team members a way to communicate about the transcription process asynchronously, directly in the DataScribe interface. Finally, it helped us plan and carry out a project workflow.

Workflow

First the project manager assigned "new" schedules (those without a transcription record) to transcribers. Transcribers then worked on the items assigned to them. If they needed to stop part way through a transcription, they could save their progress and return and finish them later; these schedules were marked "in progress." If a transcriber had a question or was unsure of an answer for the schedule they were transcribing, they could leave a note for the reviewer in the "transcriber notes" section. When a transcriber finished an item, they saved it, and then submitted it for review.

Next, the reviewers took over. They could see all the items that need review, as well as items that had been specifically assigned for them to review. If the transcription passed their review, reviewers changed the review status to “mark as approved.” If the transcription still needed more work or the reviewer had a question about what the transcriber had done, reviewers added notes to the “reviewers notes” section, and changed the review status to “mark as not approved.” This sent the transcriptions back to the original transcribers, where they could make adjustments, or complete more work on the transcription, and submit them again.

All the while, the project manager was able to use the filters built into DataScribe to monitor the number of schedules in each stage of the process: new, in-progress, submitted for review, reviewed but not approved, and approved. DataScribe also allowed the project manager to keep track of the progress of team members, looking at how many items they had, and where those items were in the workflow. This was accomplished by using the “Advanced search” option, and then selecting a team member's name under “Locked status.”

Assigning Work

Early on, we decided that it was important to assign work to individual team members. This was primarily for two reasons: (1) our project team was rather large and was working asynchronously, and (2) we had specific work priorities—i.e., certain denominations that we planned to transcribe first, based on visualizations and writing that we had planned. Assigning specific people to transcribe specific schedules helped make sure that we did not duplicate work, and that our priority schedules got transcribed first.

Before DataScribe, we used the project management software Basecamp to assign work on particular schedules to particular people. DataScribe eliminated our need for an extra piece of software, as it let us give assignments directly in the interface. For example, we wanted to research the American Rescue Workers, and create a map of their locations for our project blog, so these schedules became a priority to transcribe. DataScribe allowed our project manager to find the schedules for that denomination, mark them as a priority, and then assign them to team members who were working as transcribers. The project manager also was able to track the progress of these priority schedules, and see how many were at which stages of the transcription workflow.

Transcribers and Reviewers

DataScribe allows project managers to assign team members roles as transcribers or reviewers. (See the page in the documentation about assigning roles.) For our team, it was important to designate some people as transcribers and others as reviewers in order to keep schedules moving through our workflow. Normally we had between 6–8

transcribers and 2–4 reviewers working at one time. In DataScribe, transcribers can see items that are assigned to them, start new transcriptions, flag items with fields that seem out of the ordinary, or leave comments for the reviewers. Reviewers actually have the ability to transcribe and review; in addition to creating new transcription records, they can also see submitted transcriptions along with their flags and comments, and decide to respond to them or simply update the transcription themselves.

The project management features of DataScribe allowed the large American Religious Ecologies project team to organize, track and collaborate on the transcription process directly in the DataScribe interface; this eliminated the need for other project management software as we were able to work and communicate in DataScribe asynchronously.

Deploying DataScribe for Research

Dataset Export

Once we had gone through the labor of digitizing and transcribing the census schedules, the goal is to export the data from DataScribe so that it can be used for analysis and visualization. Exporting the data has several steps in DataScribe. It is important to note that only “approved” items are exported by DataScribe. This means that you will have to take an item all the way through the transcription process before it is included in an export. This ensures, however, that only correct and completed data is exported.

The first step to undertake in DataScribe is validating the dataset. This runs a background job in DataScribe that checks each of the items to ensure its validity. For instance, if a field on an item is supposed to be a number but is instead a string, DataScribe will flag the item for correction.

The next step is to actually export the dataset. This runs another background job on DataScribe to create the CSV file. The export process will add any new data to the CSV. Then, the CSV can be downloaded from the dataset’s page in DataScribe.

A CSV (or, comma-separated values) file is a standard way of representing tabular data. You will be able to import the CSV you download from DataScribe into many data analysis programs and languages. The CSV will contain one column, or field, for each of the fields that you create in DataScribe, and one row for each record of data you have transcribed.

You should note that, if your Omeka S items contain useful metadata, you may wish to download their metadata as a CSV as well. This can be done using the Export module for Omeka S. You can then join the two CSVs together using IDs of the Omeka items. We need to do this for the *American Religious Ecologies* project because, as previously described, we catalog the schedules with their denomination and schedule IDs as part of the process for importing them as items to Omeka.

Research and Visualizations

The main goal of the American Religious Ecologies project is to create new datasets and visualizations which provide a different way of understanding American religious history. The field of American Religious history has come a long way since the Protestant-centered denominational histories of the 1980s; in recent years the field has expanded the number and types of religious groups under study and now encompasses more of the variety of religious practice. However, not much work has been done recently in the field with datasets and visualizations. Sources which have the potential to become datasets do exist in the field; however, not many of them have been digitized, and hardly any have been transformed into datasets. In addition, some scholars have created visualizations mapping the locations of religious groups; these maps, however, have mainly focused on a single denomination, a single urban city or have their most detailed depiction of religion at the county level, instead of at the city or town level.

The American Religious Ecologies project seeks to synthesize the field of American religious history through the concept of "religious ecologies." Instead of focusing on a single religious group, or religious practice in a single location, we have been investigating how religious groups (in geographic proximity) interact with one another, and with their environment. We have been asking questions like: In any given town or city in the US, how likely was it that an individual had access to a meaningful diversity of religious options? To what degree did they have options? Did these options include non-Christian options? Were these options predominantly black or white congregations? Where were these options located? Our project is addressing these questions by creating datasets and visualizations which reveal the diversity and geographic distribution of religious groups at the level of the nation, state, county and populated place (city or town).

Creating datasets that are ready for computational analysis helps us provide a more detailed interpretation of American religion. Instead of only presenting summary statistics (counts and averages), we can look at distributions of data, and see how these changed based on factors like geography, denomination or city size. Published records from the U.S. Census of Religious Bodies have already reported on the denomination with a roughly equal number of male and female members (Latter-day Saints) or the average number of members in a Protestant Episcopal Church. Our dataset based on the 1926 Census will help us decipher the distribution of membership (or any other statistics) geographically, revealing how the number of members or gender of membership changed based on church size (small vs. large), type of of community (urban vs rural), or geographic region (Northeast, mid-Atlantic, South, west coast, midwest, etc.).

In addition, our project is adding to the field by mapping individual religious congregations from over 200 denominations at the level of the populated place (city, town or township). Other scholars have mapped religious groups at the county level, or created maps for a single city or denomination; however these types of maps make it hard to get a sense of how different denominations in immediate geographic proximity interacted with each other. By identifying and mapping the religious congregations that existed in a single city or town, we can get a better sense of this interaction.

Some of our early work had been published online in the form of interactive scholarly works—scholarly content built to take advantage of the interactive form of the web which are accessible to public audiences. These types of work showcase how our datasets and visualizations have already begun to provide new ways of thinking about American religion.

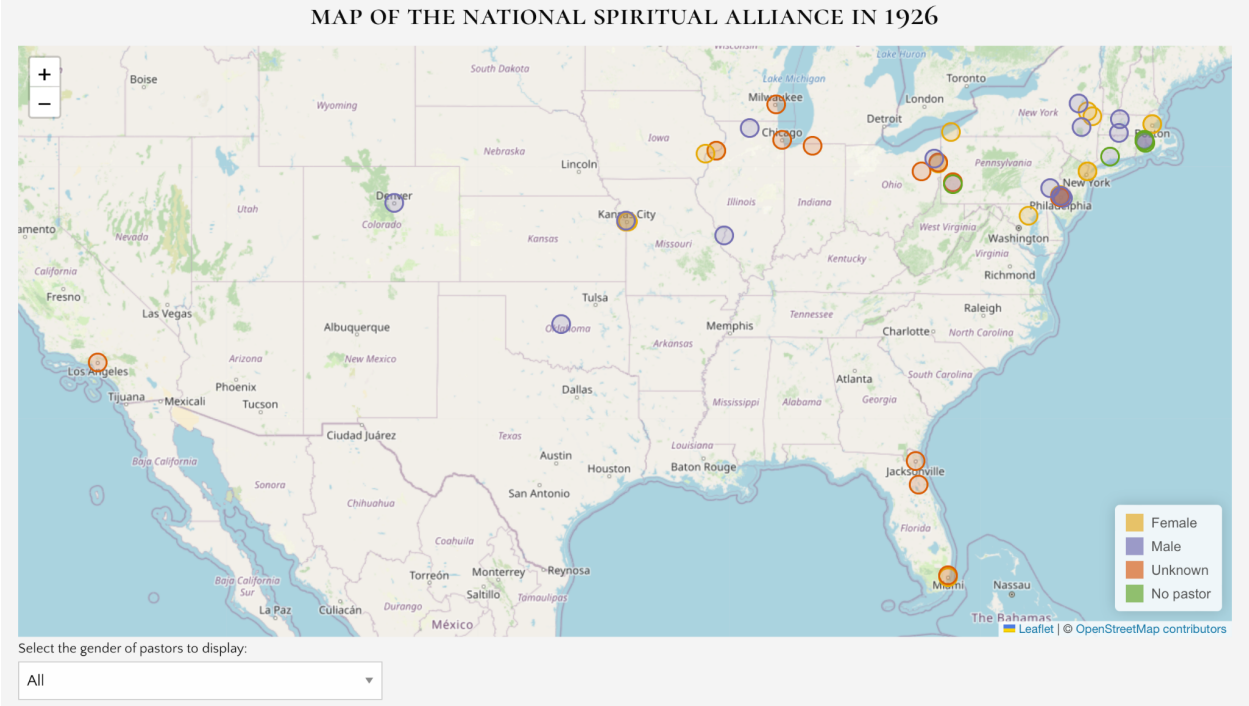


Figure 5. This map from the American Religious Ecologies project shows the location of congregations that were a part of the National Spiritual Alliance, and uses transcribed data from the 1926 Census of Religious Bodies.

Conclusion

We hope it is evident from preceding text that DataScribe has already and will continue to be a critical part of the *American Religious Ecologies* project. As a module of Omeka S, DataScribe has allowed us to seamlessly transcribe our sources in the same platform where our sources were already being stored and made available online. It has also

helped us extract qualitative information from sources and transform it into a structured dataset in one step—a process that normally takes many actions. Finally, it has increased the structure, accuracy, and usability of our data, made our datasets ready to use in computational and visual analysis, and enabled the creation of new visualizations and interpretations for the study of American religion.